can be configured to perform updates whenever a predefined number of changes have taken place in the Z-buffer used by rasterization stage 208. Asynchronous updating provides a balanced approach that retains most of the benefits of the occlusion culling method while reducing the amount of data that must be fed back from the Z-buffer. --

Please replace the paragraph beginning at page 13, line 1, with the following rewritten paragraph:

-- If the nearest Z-value of the object is less than the depth value, the object may be visible and method 400 continues at step 416. In step 416, host processor 102 performs the steps required to render the object being processed. Typically, this means that host processor takes whatever steps are required to pass the object being processed to graphics processor 112. In other cases, host processor 102 may perform the rendering process itself. --

Please replace the paragraph beginning at page 14, line 3, with the following rewritten paragraph:

-- Each coarse Z-buffer 304 in a series spans the same image 300 at a different resolution. The highest resolution coarse Z-buffer 304 splits image 300 between tiles 306 with no overlap. This is the case shown for coarse Z-buffer 304 of Figure 3. For lower resolution coarse Z-buffers 304, each tile 306 covers a larger area of image 300. Since the total number of tiles 306 remains constant, this means that tiles 306 within lower resolution Z-buffers 304 overlap each other. In fact, each succeeding lower resolution Z-buffer 304 includes successively larger tiles 306 that have successively larger overlapping areas. --

**In the Claims:**

All of the claims are being amended in this response and three new claims are added. Claims that are being amended in this response or are being added are identified with an asterisk.

*1. (Once Amended) A method for culling occluded objects from an image being rendered into a frame buffer; the method, performed by a host processor, comprising:

constructing a coarse Z-buffer, the coarse Z-buffer subdivided into a series of tiles, each tile having an associated depth value;

updating the depth values of the coarse Z-buffer using Z information from the frame buffer; and

using the depth values to selectively discard the occluded objects from the image being rendered.

*2. (Once Amended) A method as recited in claim 1, wherein updating the depth values is performed synchronously as information in the frame buffer changes.

*3. (Once Amended) A method as recited in claim 1, wherein updating the depth values is performed asynchronously.

*4. (Once Amended) A method as recited in claim 1, wherein using the depth values to selectively discard the occluded objects further comprises:

constructing a surrogate volume for an object; and

comparing nearest Z-value of the surrogate volume to the depth value of a tile that includes the surrogate volume.

*5. (Once Amended) A method as recited in claim 4, further comprising transforming the surrogate volume from object space to eye space.

*6. (Once Amended) A method as recited in claim 1, wherein using the depth values to selectively discard the occluded objects further comprises:

constructing a surrogate volume for an object; and

retrieving greatest depth value from the depth values of set of tiles that are spanned by the surrogate volume; and

comparing the nearest Z-value of the surrogate volume to the retrieved depth value.

*7. (Once Amended) A method as recited in claim 6, further comprising transforming the surrogate volume from object space to eye space.

*8. (Once Amended) A method as recited in claim 1, further comprising:

constructing a lower resolution coarse Z-buffer, the lower resolution coarse Z-buffer subdivided into a series of tiles, each tile having an associated depth value; and

updating the depth values of the lower resolution coarse Z-buffer using Z information from the frame buffer.

*9. (Once Amended) A method as recited in claim 8, wherein each tile in the lower resolution coarse Z-buffer covers the same screen area as each tile in the coarse Z-buffer.

*10. (Once Amended) A method as described in claim 8, wherein the tiles in the lower resolution coarse Z-buffer are overlapping.

*11. (New) A system, used as a host for a graphics pipeline, comprising:

a host processor executing a graphics application program, wherein the graphics application program is capable to implement:

a generation stage for creation, acquisition, and modification of information to be displayed, and organizing the information into application data structures; and

a traversal stage for traversal of the application data structures, and passing on appropriate graphics data; and

a graphics processor, communicatively coupled to the host processor, capable to implement:

a transformation stage for transformation of graphics data from object-space coordinates into eye-space coordinates, performing requested lighting operations, clipping the transformed data in clip-space, and projecting resulting coordinates into window-space;

a rasterization stage for rendering window-space primitives into a frame buffer, and performing shading calculations, texture lookups and calculations, and per-pixel operations;